

WAVELET-JARINGAN SYARAF TIRUAN UNTUK PREDIKSI DATA *TIME SERIES*

Daryono Budi Utomo

Jurusan Matematika

FMIPA Institut Teknologi Sepuluh Nopember, Surabaya

daryono@matematika.its.ac.id

Abstrak

Prediksi data *time series* dapat digunakan sebagai bahan pertimbangan dalam pengambilan keputusan yang akan datang. Jaringan syaraf tiruan merupakan metoda yang baik untuk memprediksi data *time series*, akan tetapi sulit dihindari adanya epoch (iterasi) yang banyak selama pelatihan. Sedangkan wavelet dapat dipakai untuk mendekomposisi dan merekonstruksi data sehingga dapat mengurangi banyaknya epoch. Pada makalah ini, dibahas bagaimana *Wavelet-Jaringan Syaraf Tiruan*, yang selanjutnya disebut WANN (*Wavelet-Artificial Neural Network*) digunakan untuk memprediksi data *time series*.

Ada tiga tahapan untuk mendapatkan hasil prediksi data *times series* dengan metoda WANN, yaitu *pre-processing*, *prediction*, dan *post-processing*. *Pre-processing* digunakan untuk mendekomposisi data masukan, *prediction* merupakan proses *training*, dan *post processing* dipakai untuk merekonstruksi data setelah dilakukan *training*. Selanjutnya dilakukan simulasi dengan menggunakan MATLAB. Dari simulasi ini diperoleh data *short term prediction* dan *long term prediction*.

Katakunci: *Jaringan syaraf tiruan, pre-processing, prediction, post-processing, wavelet*

1. Pendahuluan

Prediksi data time series banyak digunakan sebagai bahan pertimbangan untuk proses pengambilan keputusan, untuk itu keakuratan dalam memprediksi data memegang peranan penting agar keputusan yang diambil dapat dipertanggung jawabkan. Sebagai contoh, berapa mega watt daya listrik yang harus disediakan untuk daerah Surabaya lima tahun mendatang ? Berdasarkan data sebelumnya dapat ditentukan perkiraan kebutuhan daya listrik lima tahun mendatang, dari hasil prediksi ini dapat diambil keputusan dalam membangun jaringan listrik maupun penyediaan sumber daya listrik serta beaya yang dibutuhkan.

Ide penggunaan jaringan syaraf tiruan untuk prediksi data telah ditulis oleh [7]. Prediksi data dengan menggunakan Kalman Filter dan transformasi wavelet dilakukan oleh [6]. [4] melakukan penelitian tentang data time series untuk memprediksi stok market dengan menggunakan jaringan syaraf tiruan. [8] melakukan penelitian untuk menentukan konsentrasi NOx dan NO2 pada polusi udara dengan menggunakan metoda wavelet-jaringan syaraf tiruan. Prediksi data time series dengan menggunakan metoda gabungan wavelet dan jaringan syaraf tiruan untuk daya Listrik dilakukan [5]

Pada makalah ini akan dibahas prediksi data time series dengan menggunakan metoda Wavelet-Jaringan Syaraf Tiruan yang selanjutnya dinamakan WANN. Simulasi program ditulis dengan memanfaatkan modul-modul fungsi yang ada pada MATLAB, data aktual yang digunakan adalah data harian kurs jual-beli Rupiah terhadap dollar Amerika yang diambil dari Bank Indonesia. Untuk mengukur keakuratan hasil prediksi data digunakan Mean Absolute Percentage Error (MAPE). Model dari Jaringan syaraf Tiruan yang digunakan adalah Back Propagation satu *layer*. Jumlah *neuron hidden* yang digunakan adalah 1, 2, 3 dan 4 node, banyaknya prediktor adalah 2, 3 dan 4 data.

2. Pembahasan

Prediksi Data *Time Series* dan Wavelet

Dalam memprediksi data terdapat dua metode prediksi yaitu metode kualitatif (subjektif) dan metode kuantitatif. Metode prediksi kualitatif

lebih mengandalkan pendapat (*judgment*) dan intuisi manusia dari pada penggunaan data historis yang dimiliki[2]. Sedangkan metode prediksi kuantitatif dapat diterapkan apabila terdapat tiga kondisi berikut :

1. Tersedianya informasi tentang masa lalu,
2. Informasi tersebut dapat dikuantitatifkan dalam bentuk data numerik,
3. Diasumsikan bahwa beberapa aspek pola masa lalu akan terus berlanjut dimasa mendatang.

Tujuan dari metode prediksi data time series adalah menemukan pola dalam data time series dan mengekstrapolasikan pola tersebut ke masa depan. Prosedur untuk menemukan data prediksi untuk data times series digunakan metode moving windows artinya untuk mencari data prediksi ke- n digunakan data $(n-1), (n-2), \dots$ yang disebut sebagai prediktor. Dalam prediksi data time series terdapat dua bentuk prediksi yaitu short term prediction dan long term prediction. Short term prediction artinya untuk menemukan data prediksi menggunakan beberapa data masa lalu, sedangkan long term prediction berdasarkan data dimasa lalu dan melibatkan data hasil prediksi pada short term prediction.

Pada makalah ini, dibahas bagaimana WANN digunakan untuk memprediksi data time series. Fungsi skala ϕ dalam wavelet yang digunakan dalam makalah ini adalah fungsi skala Haar, didefinisikan oleh [1]:

$$\phi(x) = \begin{cases} 1 & ; \text{jika } 0 \leq x \leq 1 \\ 0 & ; \text{jika lainnya} \end{cases}.$$

Fungsi wavelet ψ didefinisikan oleh :

$$\psi(x) = \sum_{k \in \mathbb{Z}} (-1)^k p_{1-k} \phi(2-k)$$

untuk $z = \{0, 1\}$ dan $k = 1$ didapat wavelet Haar: $\psi(x) = \phi(2x) - \phi(2x-1)$.

Sedangkan fungsi aktivasi yang digunakan pada jaringan syaraf tiruan adalah fungsi biner yang dinyatakan sebagai berikut:

$$f(x) = \begin{cases} 1 & ; \text{untuk } \theta \geq 0 \\ 0 & ; \text{untuk } \theta < 0 \end{cases} \text{ dan } \theta = \text{threshold}.$$

Metode untuk menghitung jaringan input digunakan perkalian matriks. Jika bobot koneksi untuk *neural net* disimpan pada matriks $W = (w_{ij})$ dan input vektor x_i maka:

$$y_{in_j} = \sum_{i=1}^n x_i w_{ij}$$

Bias disertakan sebagai input dengan cara menambahkan komponen $x_0 = 1$, faktor bias diperlukan untuk training dan input bias selalu satu. Jika bobot bias dinyatakan dengan b_j maka :

$$y_{in_j} = b_j + \sum_{i=1}^n x_i w_{ij}$$

Back Propagation

Metode Back Propagation adalah metode *gradient descent* untuk meminimalkan total *mean square error* (MSE) dari output yang dihitung oleh jaringan. Training jaringan dengan back propagation meliputi tiga tahap sebagai berikut :

1. Feed forward dari pola input *training*.
2. Perhitungan error propagasi balik (backpropagation of error).
3. Pembaharuan (*adjustment*) bobot dan bias.

Setelah proses training, aplikasi jaringan hanya meliputi perhitungan *feed forward*. Arsitektur jaringan syaraf tiruan dari back propagation merupakan jaringan multi-layer. Pada unit input dan unit hidden ditambah satu node bias yang mempunyai bobot yang selalu bernilai satu. Bias pada unit output Y_k dilambangkan dengan w_{0k} , bias pada unit hidden Z_j dilambangkan dengan v_{0j} . Arsitektur jaringan multi-layer dengan satu layer, p unit hidden (unit Z) dan satu unit output (unit Y) ditunjukkan pada Gambar 1.

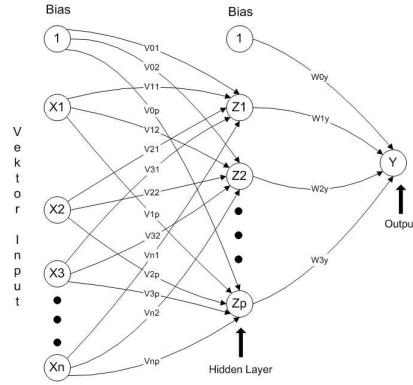
Algoritma training backpropagation[3]

Step 0 : Inisialisasi nilai bobot dengan nilai acak yang kecil.

Step 1 : Selama kondisi pada step 9 tidak terpenuhi, kerjakan step 2 - 9.

Step 2 : Untuk tiap pasangan pelatihan, kerjakan step 3-8.

Feedforward :



Gambar 1: Arsitektur Backpropagation dengan satu *hidden* layer satu output

Step 3 : Untuk setiap unit input ($X_i, i = 1, \dots, n$) menerima sinyal input x_i dan menyebarkan sinyal itu keseluruhan unit pada layer atasnya (hidden layer).

Step 4 : Untuk tiap unit dalam ($Z_j, j = 1, \dots, p$) dihitung nilai input dengan menggunakan nilai bobotnya: $z_in_j = v_{0j} + \sum_{i=1}^n x_i v_{ij}$. Selanjutnya dihitung nilai output dengan menggunakan fungsi aktivasi yang dipilih: $z_j = f(z_in_j)$.

Hasil fungsi tersebut dikirim ke semua unit pada layer di atasnya (unit output).

Step 5 : Untuk tiap unit output ($Y_k, k = 1, \dots, m$) dihitung nilai input dengan menggunakan nilai bobotnya: $y_in_k = w_{0k} + \sum_{j=1}^p z_j w_{jk}$.

Selanjutnya dihitung nilai output dengan menggunakan fungsi aktivasi: $y_k = f(y_in_k)$.

Backpropagation Of Error :

Step 6 : Untuk tiap unit output ($Y_k, k = 1, \dots, m$) menerima pola target yang bersesuaian dengan pola input dan kemudian dihitung informasi kesalahan

$$\delta_k = (t_k - y_k) f'(y_in_k)$$

Selanjutnya dihitung koreksi nilai bobot yang kemudian akan digunakan untuk memperbaharui nilai bobot $w_{jk} : \Delta w_{jk} = \alpha \delta_k z_j$.

Hitung koreksi nilai bias yang kemudian akan di-gunakan untuk mem-perbaharui nilai $w_{0k} : \Delta w_{0k} = \alpha \delta_k$, kemudian nilai δ_k dikirim ke unit pada layer sebelumnya.

Step 7 : Untuk tiap unit dalam ($Z_j, j = 1, \dots, p$) dihitung delta input yang berasal dari unit pada layer di atasnya dengan menggunakan

$$\delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk}$$

Selanjutnya nilai tersebut dikalikan dengan nilai turunan dari fungsi aktivasi untuk menghitung informasi kesalahan: $\delta_j = \delta_{in_j} f'(z_{in_j})$. Hitung koreksi nilai bobot yang kemudian digunakan untuk memperbaharui nilai $v_{ij} : \Delta v_{ij} = \alpha \delta_j x_i$ dan hitung nilai koreksi bias yang kemudian digunakan untuk memperbaharui $v_{0j} : \Delta v_{0j} = \alpha \delta_j$.

Update nilai bobot dan bias :

Step 8 : Tiap unit output ($Y_k, k = 1, \dots, m$) mengupdate bias dan bobotnya ($j = 0, \dots, p$)

$$w_{jk}(new) = w_{jk}(old) + \Delta w_{jk}$$

Tiap unit hidden ($Z_j, j = 1, \dots, p$) mengupdate bias dan bobotnya ($i = 0, \dots, n$)

$$v_{ij}(new) = v_{ij}(old) + \Delta v_{ij}$$

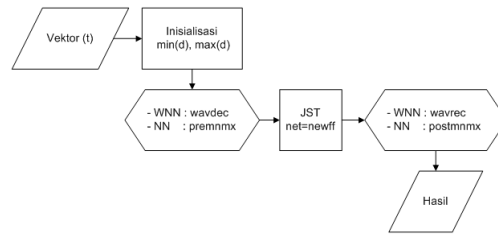
Step 9 : Menguji apakah net kondisi sudah terpenuhi.

Model prediksi

Model prediksi data time series dan struktur program serta penggunaan fungsi-fungsi ini diperlihatkan pada Gambar 2.

Untuk mendapatkan prediksi data dari model, melalui tiga tahapan sebagai berikut

1. *Pre-processing* data.
2. *Prediction*: training di Jaringan Syaraf tiruan.
3. Untuk mengembalikan data seperti semula hasil dari training dilakukan post-processing. Hasil dari post processing ini merupakan data prediksi [5]



Gambar 2: Struktur Program dan Penggunaan Fungsi MATLAB

3. Simulasi dan Uji Coba

Dalam melakukan uji coba program, menggunakan ketentuan sebagai berikut :

- Banyak data untuk training dan testing : 80
- Banyak data untuk uji short dan long term prediction: 20
- Banyak neuron yang digunakan : 1, 2, 3, 4
- Banyak prediktor yang digunakan : 2, 3 dan 4 data
- Wavelet yang digunakan : Daubechies 2

Data yang digunakan adalah data kurs beli Rupiah terhadap dolar Amerika Serikat sebanyak 120 data dari tanggal 2 Januari 2006 sampai dengan 28 Juni 2006. Untuk mengukur jaringan digunakan MAPE (*Mean Absolute Percentage Error*) yang dinyatakan dengan :

$$MAPE = \left[\frac{1}{N} \sum_{i=1}^N \frac{(x_i - y_i)}{x_i} \right] * 100\%$$

x_i = Data aktual

y_i = Data hasil training

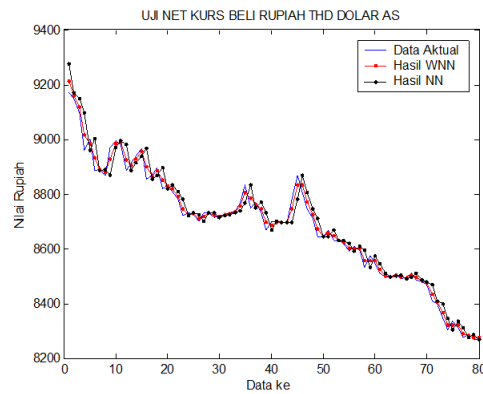
N = Banyak data yang di training.

Hasil Uji Performance Jaringan

Banyak data yang digunakan $n = 80$, hasil performance jaringan yang diukur dengan MAPE ditunjukkan pada Tabel 1. Untuk grafik hasil uji performance ditunjukkan pada Gambar 3.

Tabel 1: Performance Uji Jaringan

NH	P	MAPE WANN	MAPE NN
1	2	0.1617808 %	0.3888068 %
2	2	0.1618688 %	0.3888068 %
3	2	0.1618388 %	0.3888068 %
4	2	0.1618388 %	0.3888068 %
1	3	0.1777368 %	0.5470548 %
2	3	0.1777368 %	0.5470548 %
3	3	0.1777368 %	0.5470548 %
4	3	0.1777368 %	0.5470548 %
1	4	0.1923748 %	0.8100528 %
2	4	0.1923748 %	0.8100528 %
3	4	0.1923748 %	0.8100528 %
4	4	0.1923748 %	0.8100528 %



Gambar 3: Grafik Uji Performance Jaringan

Berdasarkan pada Tabel 1 hasil dari MAPE cukup kecil sehingga jaringan dapat digunakan untuk melakukan prediksi data.

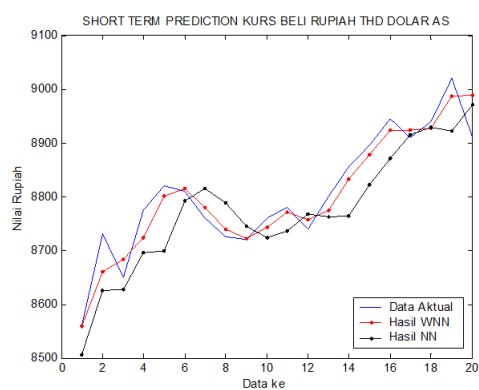
Hasil Prediksi Data Short Term Prediction

Banyak data yang digunakan $n = 20$, hasil performance prediksi da-

ta yang diukur dengan MAPE ditunjukkan pada Tabel 2. Untuk grafik hasil uji performance dengan $NH=1$ dan Prediktor =2 ditunjukkan pada Gambar 4. Berdasarkan pada Tabel 2 terlihat bahwa jika jumlah NH di-

Tabel 2: Performance short term prediction

NH	P	MAPE WANN	MAPE NN
1	2	0.2718588 %	0.6000688 %
2	2	0.2718498 %	0.7053498 %
3	2	0.2718498 %	0.7664508 %
4	2	0.2718778 %	0.6420968 %
1	3	0.3296998 %	2.1150568 %
2	3	0.3296998 %	2.3326358 %
3	3	0.3296998 %	2.1081778 %
4	3	0.3296998 %	2.6627998 %
1	4	0.3252118 %	2.6519958 %
2	4	0.3252118 %	3.1646888 %
3	4	0.3252118 %	3.0351488 %
4	4	0.3252118 %	2.9289798 %



Gambar 4: Hasil Short Term Prediction $NH = 1$ Prediktor = 2

tambah, hasil MAPE tidak berkurang. Jadi penambahan jumlah NH tidak memperbaiki hasil prediksi data. Penambahan pada banyak prediktor hasil MAPE semakin besar, jadi penambahan jumlah predictor mengakibatkan menurun-nya hasil prediksi data. Berdasarkan grafik pada Gambar 4, hasil prediksi data WANN maupun NN sudah mengikuti tren data aktual sehingga dapat dikatakan bahwa hasil prediksi sudah sesuai.

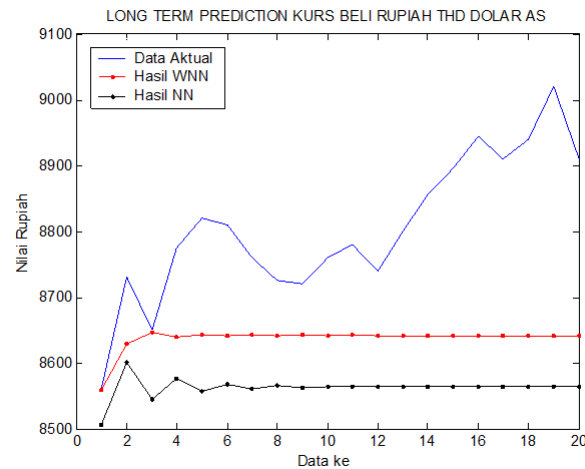
Hasil Prediksi Data Long Term Prediction

Prediksi data ke-1 pada long term prediction masih merupakan hasil prediksi data short term prediction. Hasil prediksi data ke-2 dan seterusnya merupakan hasil dari long term prediction. Performance untuk long term prediction diperlihatkan pada Tabel 3 dan tampilan grafik untuk hasil dapat dilihat pada Gambar 5.

Tabel 3: Performance Long Term Prediction dengan $n = 20$

NH	P	MAPE WANN	MAPE NN
1	2	1.8919328 %	2.6419668 %
2	2	1.8919328 %	2.4447918 %
3	2	1.8919328 %	4.0309788 %
4	2	1.8919328 %	2.4576868 %
1	3	2.9932258 %	3.6277098 %
2	3	2.9932258 %	5.5225828 %
3	3	2.9932258 %	3.3162208 %
4	3	2.9932258 %	7.5679428 %
1	4	4.2723738 %	2.1618958 %
2	4	4.2723738 %	4.1687428 %
3	4	4.2723738 %	4.0560008 %
4	4	4.2723738 %	3.0914318 %

Pada kasus long term prediction, WANN maupun NN hanya mampu sampai prediksi data yang ke-3, setelah itu hasil prediksi data menuju ke suatu nilai yang sama sehingga hasil prediksi data sudah tidak layak digunakan. Hal ini disebabkan selisih dari prediktor terlalu kecil dan menuju ke suatu nilai 0, sehingga hasil dari prediksi data menuju ke suatu nilai yang



Gambar 5: Grafik Hasil Long Term Prediction $NH = 1$, Prediktor = 2

sama yang terlihat pada Gambar 5 menunjukkan grafik hasil yang sudah tidak dapat ditoleransi.

4. Kesimpulan dan Saran

4.1. Kesimpulan

Dari hasil uji coba program yang telah dilakukan diperoleh kesimpulan :

- Jumlah node pada neuron *hidden* tidak dapat meningkatkan performance jaringan WANN maupun Pre-NN.
- Banyak prediktor yang digunakan mempengaruhi hasil prediksi data, semakin banyak prediktor hasil prediksi data semakin tidak akurat.
- MAPE hasil dari prediksi data WANN lebih kecil dibandingkan dengan Pre-NN, dengan demikian hasil prediksi data WANN lebih baik dibandingkan dengan Pre-NN.
- Prediksi data dengan WANN menghasilkan nilai MAPE yang sama sehingga dapat dikatakan metode ini stabil.
- WANN maupun Pre-NN hanya mampu untuk prediksi data short term prediction.

4.2. Saran

Untuk penelitian selanjutnya dapat dikaji banyak data minimal untuk training, pre-processing dengan menggunakan bentuk-bentuk wavelet yang lain misal, Coiflet Wavelets, Symlet Wavelets

Pustaka

- [1] BURRUS C. SIDNEY, DKK., *Introduction to Wavelets and Wavelet Transforms*, Prentice Hall International, Inc, Housto, 1998.
- [2] CRYER, JONATAN D., *time series Analysis*, PWS-KENT Publishing company, Boston, 1996.
- [3] FAUSETT LAURENE, *Fundamentals of Neural Network*, Prentice Hall, New Jersey, 1994.
- [4] LIN FENG, DKK, *Times Series Forecasting with Neural Network*, Central Queensland Universty, Australia, 1994.
- [5] LOH RUEY HWA, *Times Series Forecast with neural network and wavelet techniques*, The University of Queeland, 2003.
- [6] REANAUD OLIVER DKK, *Kalman-type Filtering using the Wavelet Transform*, Elsivier Science, 2000.
- [7] VLAD SORIN, *On the prediction methods using neural network*, University of Suceava, Romania, 2002.
- [8] ZHANG ZHGUO, SAN YE, *Adaptive Wavelet Neural Network for Prediction of Hourly NO_x and NO₂ Concentration*.